

The Decomposition of HPEC Applications Mapped to The Natural Decomposition of a Solution Architectures – Another Way to Think About Solving HPEC Problems

Joe Germann
SKY Computers

Email: <mailto:germann@skycomputers.com>

Abstract: In the HPEC historical past, all large-scale computer architectures that were designed, developed, and delivered by the so-called DSP vendors generally were of the “kitchen sink” design approach. Each board unit basically incorporated all of the features that one could possibly want into a single unified design. Processors, I/O, memory, and interconnects were built in whether they were needed or not. They were all available in a single unified product thereby increasing per slot functionality but many times at the burden of complexity, reliability, and overall cost of ownership. While this was convenient and appropriate at the time, the user was required to procure and support all of the key building blocks with solutions that were, for the most part, vendor unique and quite frankly not always necessary for application success.

Through the examination of high performance, high end embedded computing applications, it can be shown that nearly all of them have a very distinctive and natural decomposition of the problem space such that there lies a distinct I/O and data management portion and another distinct compute portion. Wrapped around both components is the need for global (infrastructure wide) communications as well as system state and health management. We further learned that in most instances that the delivered solution was burdened with many facilities that were simply not of any value and in many cases robbed the system of valuable real estate and/or reliability simplicities.

The onslaught of blade servers has caught the eye of many a developer with the *apparent* low cost of ownership, but upon close examination, they too suffer from feature bloat for most HPEC applications. However; the sheer simplicity and ease of use in small form factor blade servers are attractive for many needs that don’t *really* require harsh, dense packaging but not in the real time Signal and Image Processing arenas that High Performance Embedded Computers are warranted. Right form factor, right price, wrong feature set for HPEC needs.

This paper will address an industry unique architectural approach to addressing the needs of HPEC applications through the use of distinctive, upgradeable, and naturally decomposed solution elements. This approach provides an application space with the freedom to address complex solutions needs without the burden of the “kitchen sink” approach. The focus of this talk is on the adoption of highly flexible Data Acquisition Servers and highly focused Compute Servers, each of which addresses the unique and very demanding needed of real time signal and image processing. Advances in COTS components (hardware and software) are cornerstones of this capability and future flexibility.

A comparison between the more traditional HPEC systems and the Next Generation Architectures will be presented for key applications within both the Defense and Commercial communities.

Report Documentation Page				Form Approved OMB No. 0704-0188	
Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.					
1. REPORT DATE 20 AUG 2004		2. REPORT TYPE N/A		3. DATES COVERED -	
4. TITLE AND SUBTITLE The Decomposition of HPEC Applications Mapped to The Natural Decomposition of a Solution Architectures Another Way to Think About Solving HPEC Problems				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) SKY Computers				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release, distribution unlimited					
13. SUPPLEMENTARY NOTES See also ADM001694, HPEC-6-Vol 1 ESC-TR-2003-081; High Performance Embedded Computing (HPEC) Workshop(7th). , The original document contains color images.					
14. ABSTRACT					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT UU	18. NUMBER OF PAGES 13	19a. NAME OF RESPONSIBLE PERSON
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified			



***The Decomposition of HPEC Applications Mapped to
the Natural Decomposition of a Solution Architecture***

Historical Solution Drivers for HPEC Applications

❑ *Circa 1980 - 1990*

- Specialized Co-Processors Tightly Coupled to the Host CPU
- Dual-Ported Memories; albeit very Small Amounts
- Data Acquisition via Host CPU Bus

❑ *Circa 1990 - 1995*

- First Wave of Small-Count Multi-Processor Embedded Applications
- Localized Non-Shared Memory
- Low Bandwidth, Low Functionality Bus-Based Interconnects
- Data Acquisition via Direct Parallel I/O Ports on Each Card

❑ *Circa 1995 - 2003*

- First Wave of Larger-Count Multi-Processor Applications
- High Bandwidth, Crossbar Based Interconnects
- Lots of Processors and Lots of Small Distributed Memory Pieces
- Data Acquisition via Direct Fabric Based Ports on Each Card
- The era of the Compact Application Benchmark

Historical Trends for Application Fitment to Hardware

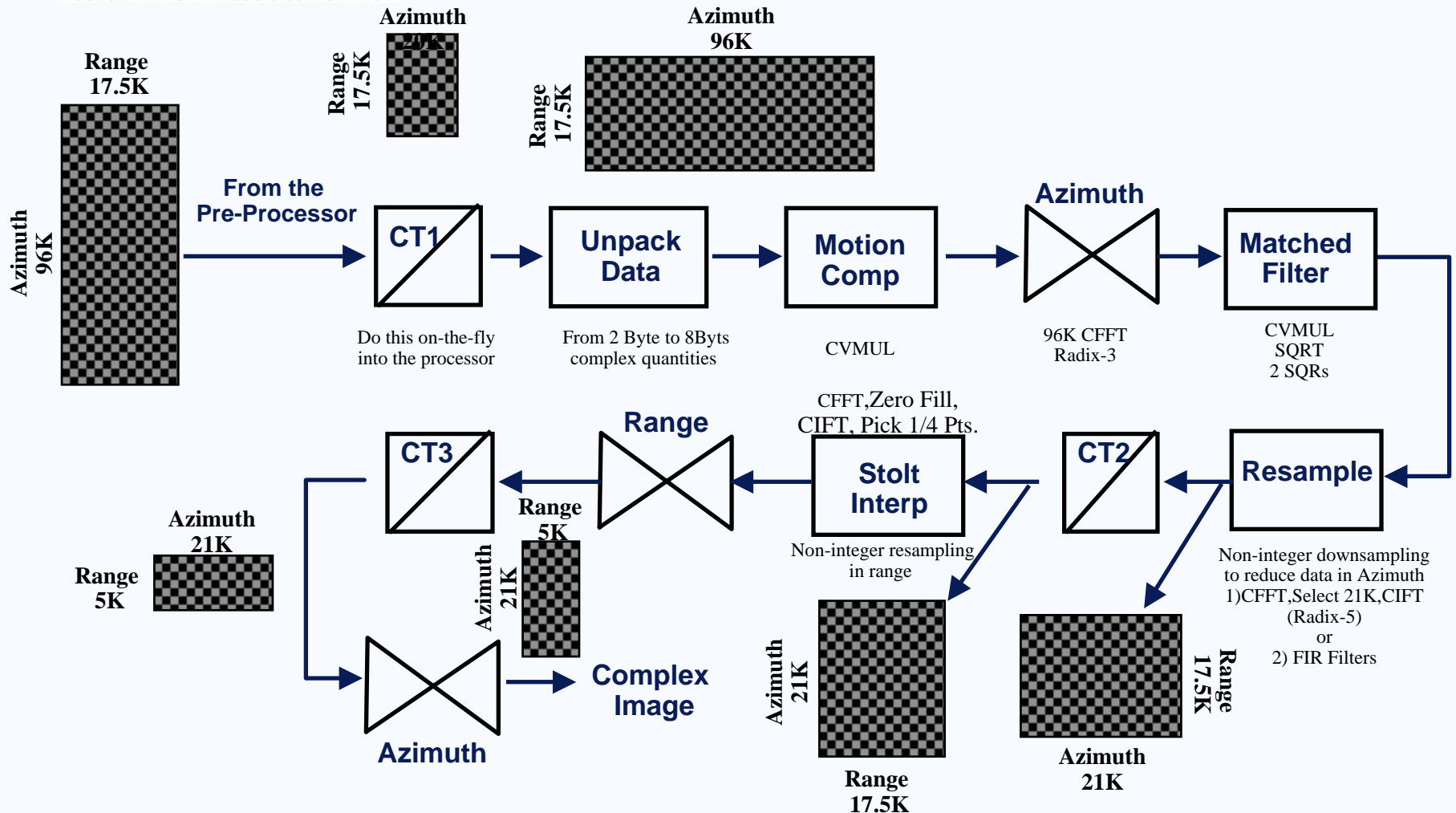
❑ It was

- Finesse the I/O *and* Computations to Fit the Computer's Architecture which included Integrated I/O and Computations
- Buffer, Rearrange, Move, and Process the Data *In-Fabric* using Widely-Distributed Small Buffers of Memory
- Tightly Couple the Application's Architecture to the Distributed Memory and Computer's Architecture

❑ The Community had an Approach to Understand Behavior and Performance Estimates for Complex HPEC Systems

- Standard Benchmarks: 2DFFT, Corner Turns, STAP, etc
- Help it with Middleware: MPI/RT, Data Re-Org, VSIPL
- But they did not really address the I/O and the impacts thereto to the Overall Processing and Processing Management of Data

The FOPEN Example



Note 1: To Preserve Memory pack and unpack the data before and after corner turns

Note 2: To Preserve and Bandwidth Convert to/from float and Integer

The Architecture Problem

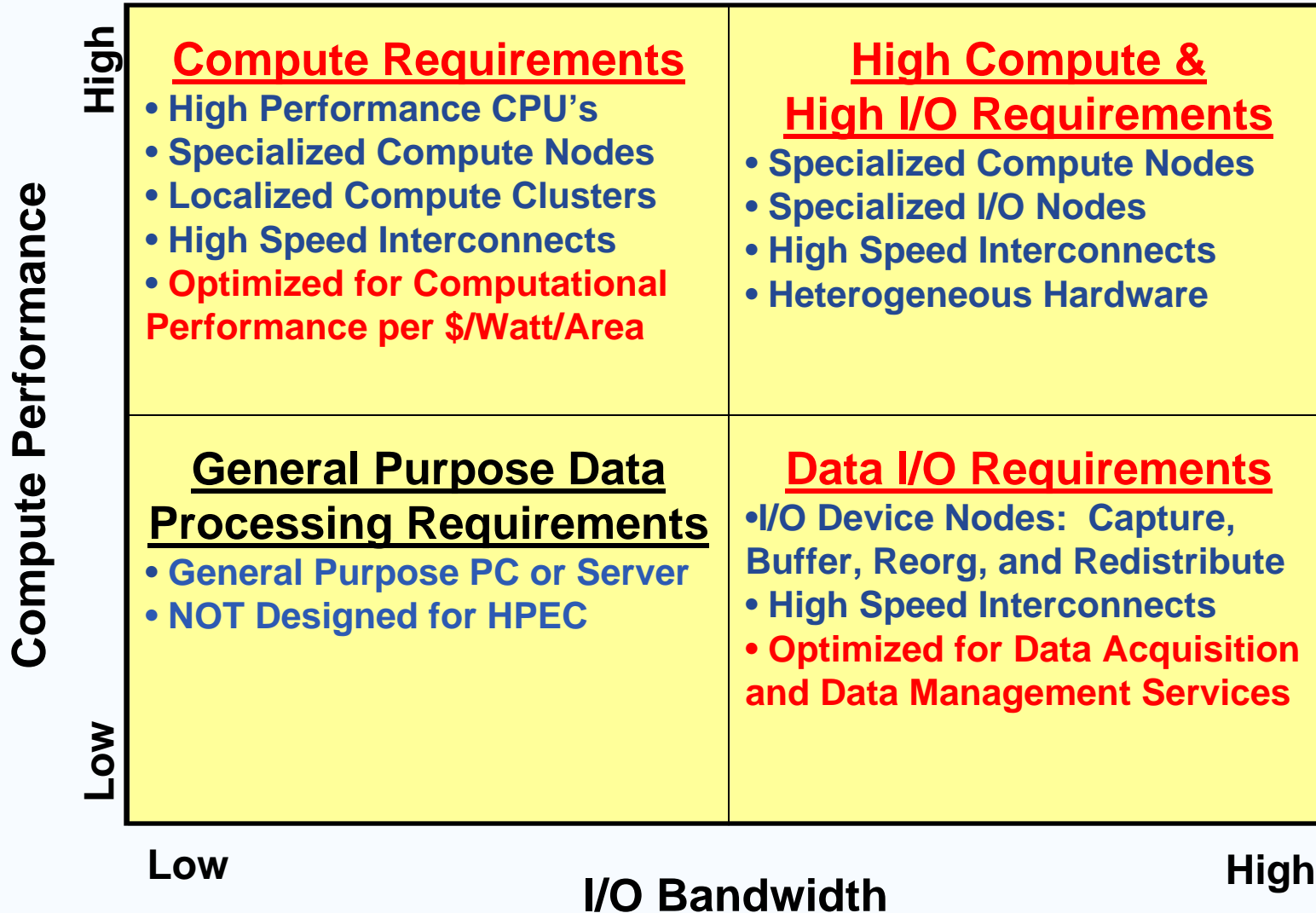
- ❑ ***Small Chunks of Distributed Memory can be Problematic***
 - Lots of Small Memory Buckets = Lots of Data Movement
 - Lots of Data Movement = Lots of Bandwidth Needed
 - Hence the Problem: The Memory becomes *in-fabric*

- ❑ ***Small Memory Buckets are Challenging to Manage***
 - Data Feeds are Scattered among the Fabric
 - Partial Data Sets are Unnaturally Broken Up
 - Many times, way too Much Scatter, Gather, and Re Organization

- ❑ ***HPEC Problems Naturally Decompose into Two Key Areas***
 1. Data Acquisition, Buffering, and Re-distribution
 2. High Speed and Highly Complex Computations on Well Bounded Data Sets utilizing Well Bounded Algorithms

They have very different Architectural Needs if they are to be Optimally Served

HPEC Application Needs Mapping



What's Different in Today's Computer Architectures

- ❑ ***Processor-to-Memory Bandwidth is Huge***
 - 64-bit Wide DDR; > 3 GB/Sec memory access speed is possible
 - Memory is Cheap and Abundant
- ❑ ***I/O and Local System Bus Bandwidth is Very High***
 - Commodity Busses e.g. PCI-X > 1GB/Sec
 - Lots of Peripherals, Lots of Available Software (driver) Support
- ❑ ***Interconnect Fabrics are FAST and SMART***
 - 1GB/Sec per Port
 - Self-Discovery, Fault Detection, Recovery, Reliable
- ❑ ***Standard Processors are Readily Available; Specialized Devices are Becoming Easier to Use***
 - High Throughput SIMD's, DSP's, Fast Server-Centric Processors
 - FPGA's, ASIC's, and Other Custom Logic
- ❑ ***Today, it is Easier to Balance I/O and Computational Needs at the Computer Level rather than at the Application Level***

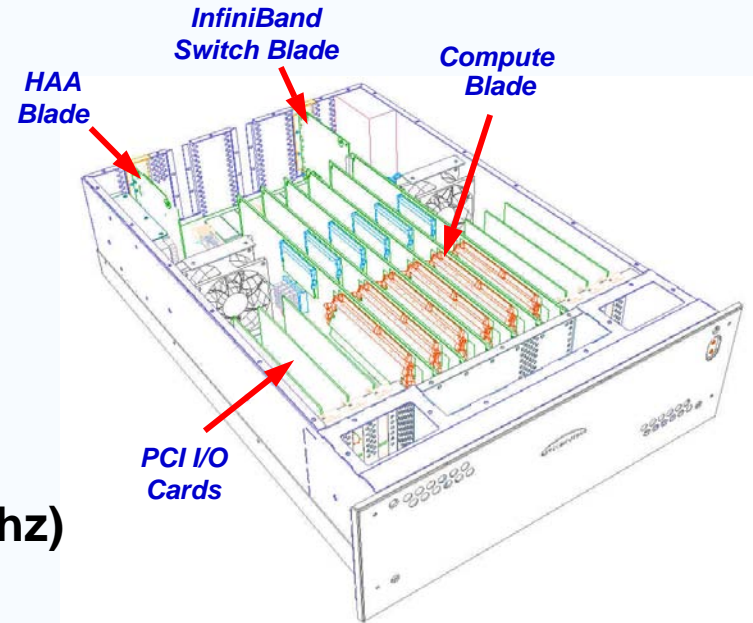
Data Acquisition Server

SMARTpac 600



**Optimized for Data Acquisition and
Data Management Services**

- ❑ **Six PCI-X adapter slots
(PCI 32-bit/33Mhz up to PCI-X 64 bit/100Mhz)**
- ❑ **Six compute blade slots for single or
dual-processor compute blades**
- ❑ **Six InfiniBand connections at rear of chassis**
- ❑ **Two RJ45 Ethernet connections for High Application
Availability Infrastructure**



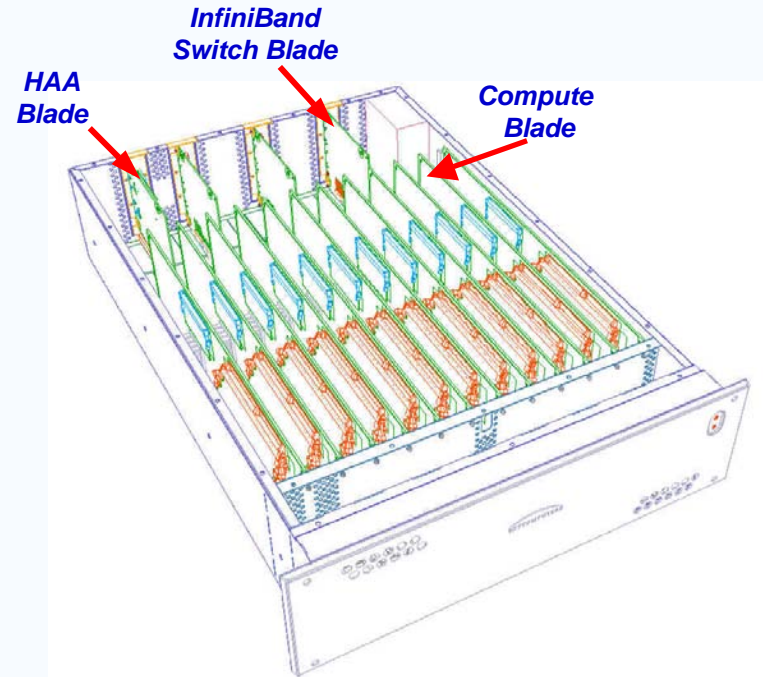
Compute Server

SMARTpac 1200



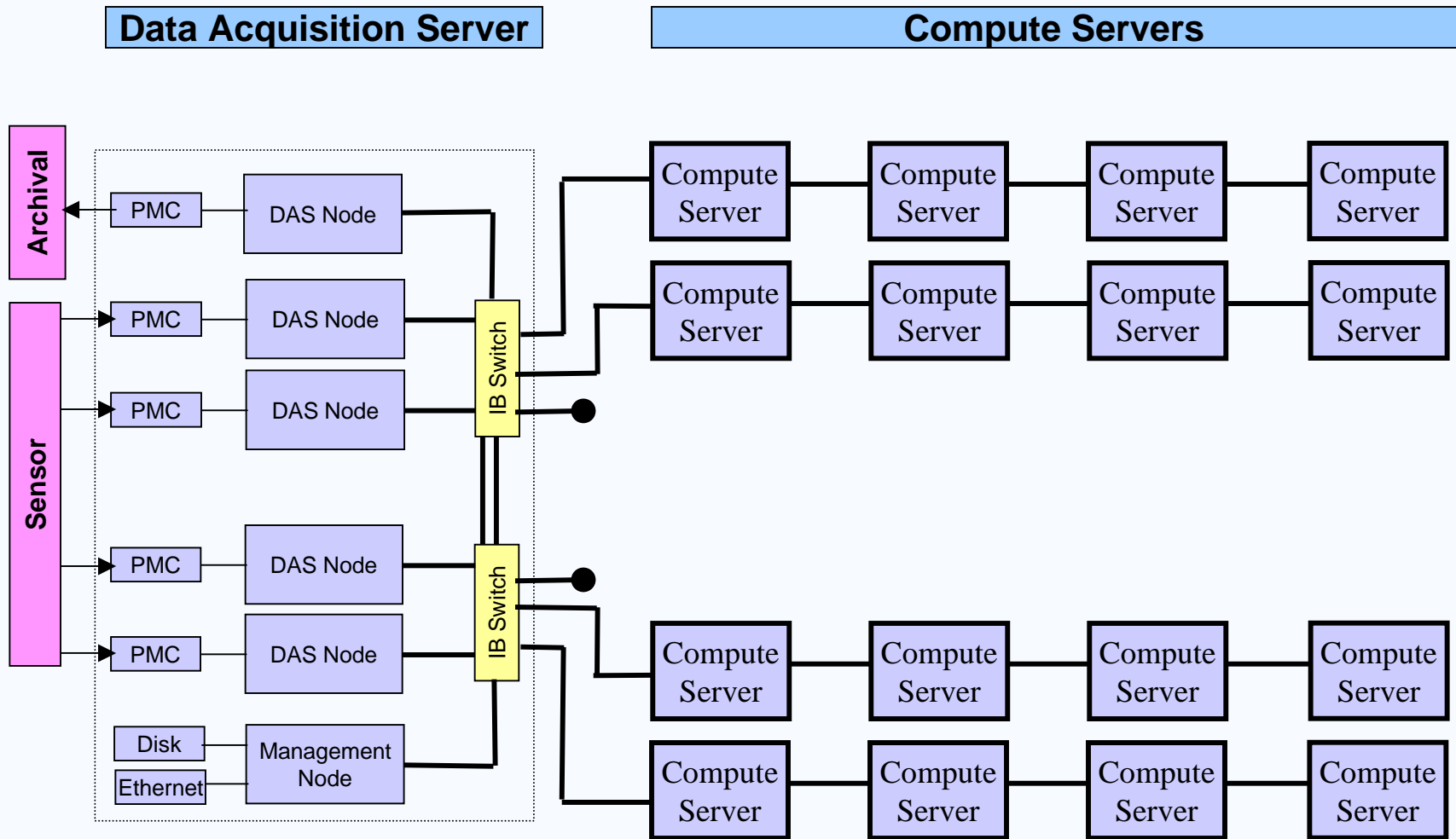
**Optimized for Computational
Performance per \$/Watt/Area**

- ❑ Twelve compute blade slots for single, dual-processor, or special function compute blades
- ❑ Six InfiniBand connections at rear of chassis
- ❑ Two RJ45 Ethernet connections for High Application Availability Infrastructure



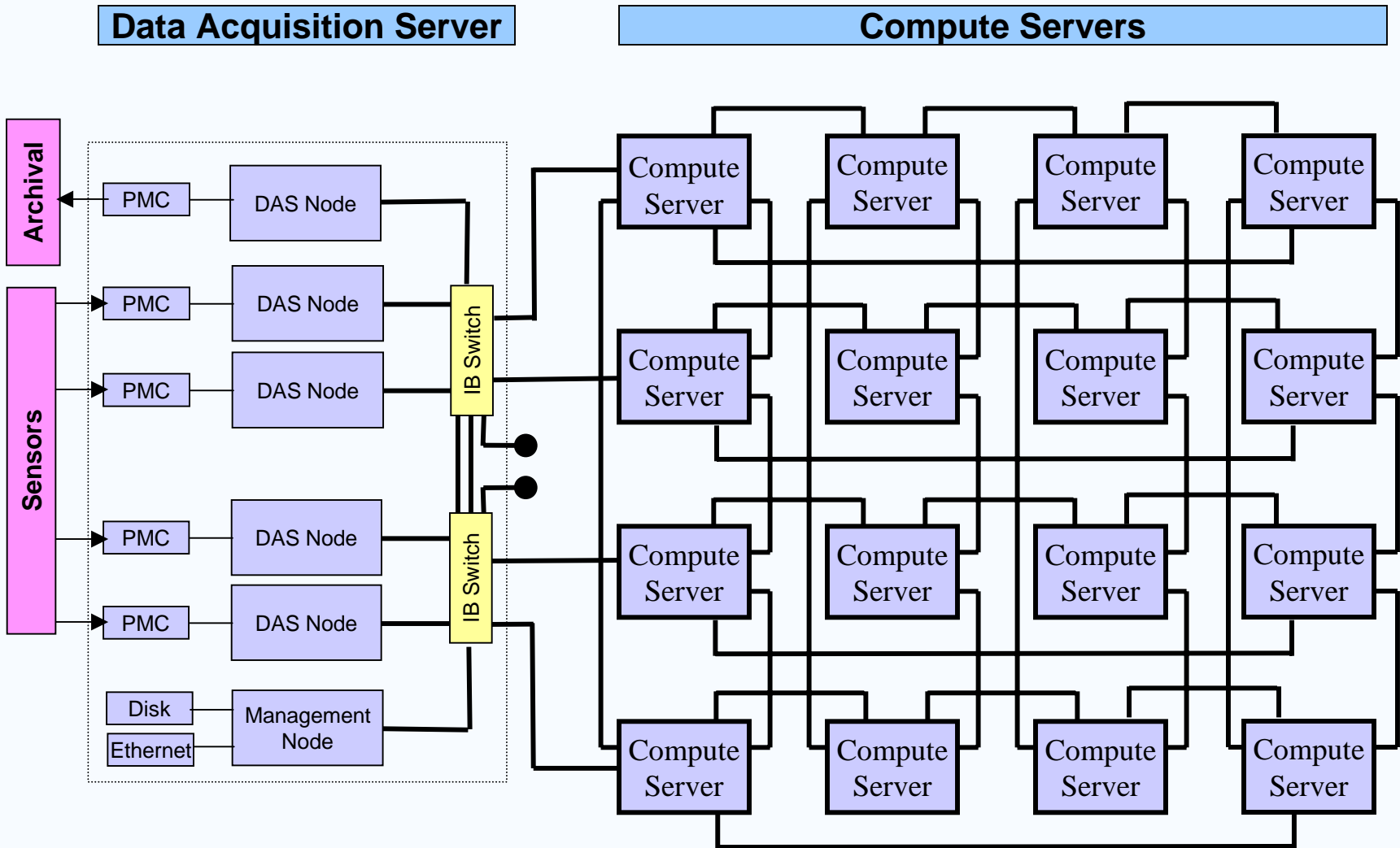
Piped Connection

Medium Bandwidth – Lower Complexity



Fully Connection Mesh

High Bandwidth – Higher Complexity



Decomposition Summary

- ❑ **Most HPEC Problems Naturally Decompose into:**
 1. Data Acquisition and Management Services, and
 2. Computational Services

- ❑ **The Current HPEC Systems built around Raceway™, SKYchannel™, and Myranet™ are representative of an “older school” approach, whereby I/O and Computes are tightly coupled, physically bound together, and use Small Buckets of Fabric-based Shared memory**

- ❑ **Today’s Technologies allow one to Think and Actually Implement Differently to Meet an Application’s Actual Decomposed I/O and Processing Needs**
 - **Data Acquisition Servers** optimized for I/O Services, Data Buffering, Data Management, and Data Distribution
 - **Compute Servers** optimized for Signal and Image Processing